EPJ Web of Conferences **360**, 01010 (2026) https://doi.org/10.1051/epjconf/202636001010
QUICK'26

An Experimental Analysis on the Application of Quantum Federated Learning in Healthcare Analytics with Variational Parameter Set-up

Jeya Mala Dharmalingam[1]*, *Anto Cordelia Tanislaus Antony* Dhanapal[2], *Santhosh D*[1]
[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India.
[2]Department of Chemical Science, Universiti Tunku Abdul Rahman (UTAR), Malaysia

Abstract. The paradigm shift of moving from traditional machine learning models to distributed learning using Federated learning has gained its importance in various real-time applications especially in applications that requires privacy preservation as the primary criterion. However, for complex real-time data, the computational limitations and communication bottlenecks in the current federated learning environment has made the advancements of applying Quantum computing in Federated Learning. Quantum Federated Learning (QFL) is combining FL with Quantum computing. In this paper, QFL is applied for a real-time healthcare analytics dataset to assess the applicability of Quantum computing in such cases. From our implementation, we realized that, even though QFL has several advantages in terms of distributed model training with quantum-based communications, it suffers from accuracy concerns as the current implementation is experimented in the classical system configuration and not in the QPU or Quantum computer set-up. We concluded that, if the Quantum computers or QPU are utilized for such complex real-time dataset, we can get the real advancements QFL.

1 Introduction

Quantum Federated Learning (QFL) combines the characteristics of decentralized training of Federated Learning (FL) with Quantum computing. The training will be happened at each client side and based on that, the model updates are communicated to the Quantum Machine Learning components. Hence, the local models which are deployed at the client sides will handle the local data and parameter updates will be used in the global aggregation with high level of security and privacy preservation. QFL has the properties of faster computation as the Quantum Processors can handle complex models with high dimensional data than the classical processors. This reduces the training time required for the models. By means of the Quantum Key Distribution (QKD) techniques and Quantum Encryption we can secure the data communication in a secured manner. It also helps in resource sharing as the resources are optimized to help the clients with limited computational capability by means of distributed processing across the clients.

---

* Corresponding author: jeyamala.d@vit.ac.in

The proposed work focuses on addressing the issues in the existing solutions such as: Centralized data architectures that are incapable of providing parallel access on large medical records across the state; access issues on data privacy concerns across hospitals, non-generalization of traditional machine learning models; security gaps in applying the current cryptography techniques.

Based on the observations as indicated, the proposed work applies the transformational scientific approach. This work provides a detailed experimental analysis of the impact of Quantum Federated Learning in real-time healthcare analytics with varying client types and variational parameters setup.

2. Literature Survey

One of the most comprehensive contribution is from Ren et.al [1] on the challenges related to quantum hardware scalability, noise vulnerability and deployment feasibility in applying QFL in predictive data analytics. They discussed two branches namely Efficiency-oriented QFL (EffQFL) and Security-oriented QFL (SecQFL).

Yu et.al [2] has introduced a multi-party computation technique on quantum gradient descent methods. They focused on quantum communication networks with scalable, privacy – preserved architecture. They provided a privacy-aware model over high-dimensional datasets.

Li et.al [3] has demonstrated the application of QFL as a privacy-preserving framework. It prevents quantum servers and third particies from accessing client-sensitive data.

His protocol incorporated Variational Quantum Classifiers with privacy preservation mechanisms.

The framework proposed by Song et.al [4], has combined classical Edge Clients with centralized quantum servers using shadow tomography for inferencing without exposing client-side data. They tested it on MNIST dataset and provided a scalable bridge between classical and quantum resources.

In a work by Rofougaran et.al [5] has applied QFL with privacy preservation on image classification datasets and achieved more than 98% accuracy with privacy preservation. The integration of Federated Learning (FL) with Quantum has improved privacy with quantum advantage.

Shi et.al [6] has proposed a methodology to handle data heterogeneity issue in Federated systems using QFL. The QFL model dynamically adjusts quantum layers in the client level. The solution has provided a unique model with shared global model by improving accuracy.

The first QFL model proposed by Chehimi et.al [7] has incorporated Quantum circuits with LSTM which they termed as 'FedQLSTM'. It was designed to handle function approximation tasks with remarkable improvements in training speed and convergence.

Innan et.al [8] has made improvements in FedQNN by leveraging the QNN for complex Genomics analytics. The model has achieved more than 86% accuracy with reduction in communication costs.

A QFL framework introduced by Wang et.al [9] has applied ring topology based quantum networks with quantum weight updates and non-transition from classical to quantum states. This has reduced latency and improved deployment in quantum network environments.

A personalized Quantum Federated Learning (QFL) was proposed by Gurung et.al [10] considering Euclidean distance with weighted personalization by means of quantum federated averaging. Their method dynamically adopts to data heterogeneity, improved performance and was validated using theoretical and empirical analysis.

3. Proposed Methodology

The research objective of this proposed work focuses on developing a Quantum Machine Learning (QML) based Federated Learning (FL) (QFL) framework tailored for disease diagnosis across distributed healthcare systems from different hospitals. The primary goal is to enhance data privacy, security, and diagnostic accuracy by leveraging the advantages of both federated learning and quantum computing. The proposed methodology's mathematical formulations are given below:

The global optimization function is defined in Eqn.(1)

In QFL, multiple clients collaboratively minimize the global loss, hence the optimization function is to minimize the global loss as indicated in Eqn.(1).

$$F(w) = \Sigma \frac{n_k}{n} F_k(w) \qquad (1)$$

Where, w is the global model parameters which are the quantum circuit parameters; K is the number of clients; nk is the data samples at client k; n is the total number of samples; Fk(w) is the local loss function.

The local Quantum Gradient is calculated for each client using quantum circuits using Eqn. (2).

$$\mathrm{g_k} = \nabla F_k(\mathrm{w_t}) \qquad (2)$$

Where, gk is the gradient of local quantum model and wt is the set of global parameters at round t.

The individual local models update will be done using the formula given in Eqn. (3)

$$\mathrm{w^k}_{\mathrm{t+1}}\mathrm{k} = \mathrm{w_t} - \eta \mathrm{g_k} \qquad (3)$$

Where η is the learning rate; $\mathrm{w^k}_{\mathrm{t+1}}$ is the updated parameters at client k

Here, global aggregation is done using federated averaging. It is given in equation 4.

$$\mathrm{w_{t+1}} = \mathrm{w_t} - \eta \sum_{\mathrm{k=1}}^{\mathrm{k}} \frac{\mathrm{nk}}{n}\, \mathrm{gk} \qquad (4)$$

For variational quantum circuits in quantum, the formula given in Eqn. (5) is used.

$$\frac{\partial f(\theta)}{\partial \theta} = \frac{\left(f\left(\theta+\frac{\pi}{2}\right)-f\left(\theta-\frac{\pi}{2}\right)\right)}{2} \qquad (5)$$

Where, f(θ) is the expectation value of quantum circuit and θ is quantum gate parameter.

The proposed methodology has the following phases:

Phase 1 : Federated Learning Model Development with Dataset collection will be done from existing records availability.

Phase 2: Implementation of Quantum Machine Learning (QML) with Federated Learning Strategy.

The entire workflow is going to be implemented in three stages:

Stage 1: Health record Data Pre-processing for Quantum Compatibility

In the first stage, we transform heterogeneous clinical and diagnostic data such as laboratory values, medical imaging metrics, and sensor readings into appropriate feature representations using suitable pre-processing techniques depending on the type of the clinical modality. This step is crucial for applying Quantum Neural Network (QNN) based Quantum Machine learning, which requires curated feature representations as input to effectively process data within quantum circuits. By converting numerical health records into structured data, we enable the quantum models to extract intricate patterns and features relevant to various diseases.

Stage 2: Application of Quantum Machine Learning for Disease Classification

In this phase, as shown in figure 1, we employ advanced QML algorithms, including Quantum Neural Networks (QNNs) to analyze and classify disease-specific datasets ranging from cardiovascular conditions to neurological disorders. These algorithms utilize the principles of quantum superposition and entanglement to perform more efficient and potentially more accurate classification than classical counterparts as illustrated in Figure below.

Our aim is to enhance diagnostic precision and uncover subtle biomarkers that might be missed by traditional machine learning approaches.

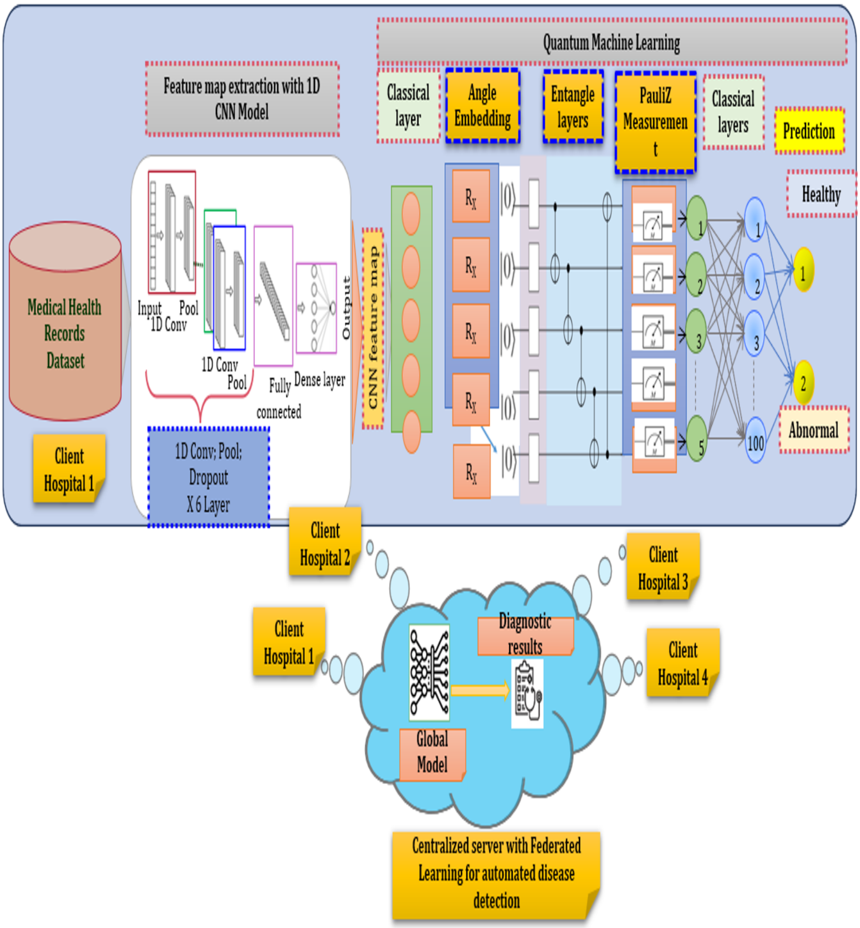

Fig 1. Individual Privacy preserving Quantum Machine Learning Model for Diagnosis in Hospital Environment

Stage 3: Federated Learning with Quantum Model Updates

The final stage integrates trained quantum models into a Federated Learning environment using the Federated Averaging (FedAvg) technique as illustrated in Figure below. Each participating healthcare node (e.g., hospitals, diagnostic labs, wearable health devices) trains the quantum model locally using its private patient data. Only quantum-derived model weights and gradients are shared with a central aggregator not the raw data. This approach ensures strong data privacy and regulatory compliance while supporting collaborative learning across institutions. By incorporating QML into FL, we aim to develop a scalable and secure diagnostic framework capable of learning from diverse, real-world medical data without compromising patient confidentiality.

4. Experimentation and Result Analysis

For a preliminary study, the Telengana State health department patients' diagnostics dataset (Open Data Telengana) is taken from Govt. of India's (India.AI) website: https://data.telangana.gov.in/dataset/telangana-health-department-diagnostics-data [11].

QFL basic working is experimented on sample dataset, we need to proceed with the above open dataset with further hyper parameter fine-tuning.

The experimentation is done in the following experimentation set-up given in table 1.

Table 1. Experimentation Set-up

Hardware Configuration	Software Configuration
HP 255-G9 Notebook PC, 1TB HDD, 4GB RAM (extendable upto 64GB), AMD Athlon Silver 3050U with Radeon Graphics (2.30 GHz), 64-bit OS	64-bit operating system, Colab with TPU run-time environment, Python (Pennylane, Tensorflow, sklearn etc.)

4.1 Initial set-up

From pennylane, import qml for quantum machine learning model creation. As the dataset is reduced to 4 features based on the features importance, the number of qbits is set to 4. Then the quantum circuit is created with the inputs and the weights associated with each of them.

The Quantum layers are created with angle embedding and quantum entanglement for measurement.

Step 1:

Load the PCA dataset (after applying Principle Component Analysis, only the relevant features are retained from the high-dimensional dataset).

```
Train shape: (131368, 4)
Test shape: (131368,1)
Label distribution in train:
label
1   81
0   79
Name: count, dtype: int64
```

```
Label distribution in test:
label
0    20
1    20
Name: count, dtype: int64
```

Step 2:

Setup the quantum layers with QNN (Quantum Neural Network) with number of qbits as the number of features (4).

Step 3:

Create the QNN model with layers of activation functions as Relu and sigmoid and optimizer as ‘Adam Optimizer’, loss function as ‘cross entropy’ and the evaluation metric as ‘accuracy’.

4.2 Implementation of Hybrid Clients Model

Step 4:

For experimentation, the number of clients is set-up as 5. Among them, the ones which are quantum enabled and classical can be set-up. In this work, two clients are set-up as quantum enabled.

```
[[0.48203294 0.13501349 0.21256882 0.87060908]
 [0.51432247 0.64515525 0.77229744 0.17213658]
 [0.06012114 0.76715348 0.21773372 0.17880344]
 [0.6352395  0.87446082 0.3586487  0.50497896]
 [0.09843757 0.74551759 0.33765244 0.56415333]
 [0.53909472 0.45688042 0.69352234 0.21776357]
 [0.61183594 0.39044277 0.08264054 0.30437533]
 [0.71851693 0.39074338 0.04178229 0.59299599]
 [0.5875233  0.1195895  0.6104805  0.77048785]
 [0.40012427 0.7133638  0.49007107 0.49675079]
 [0.13347082 0.34635506 0.83425008 0.5932557 ]
 [0.69527349 0.82983959 0.96108649 0.28034024]
 [0.42574679 0.47932052 0.1121575  0.09401855]
 [0.25143327 0.50810295 0.39798209 0.9721102 ]
 [0.45787738 0.26546004 0.70306621 0.52250949]
 [0.53821785 0.99702174 0.23844453 0.13112246]
 [0.75926542 0.466431   0.1140802  0.36900275]
 [0.57058298 0.28714765 0.97578687 0.29001802]
 [0.17157469 0.22235597 0.46467921 0.33268562]
 [0.96573777 0.45847322 0.01676637 0.22116624]
 [0.71080511 0.56471923 0.44507021 0.99316206]
 [0.9349983  0.46345805 0.42773941 0.20848533]
 [0.23282524 0.72686305 0.50622941 0.91115693]
```

```
 [0.80754532 0.74062706 0.04354293 0.06321484]
 [0.01396396 0.98665407 0.79812897 0.48151744]
...
 [0.93651496 0.48315224 0.42089702 0.93226469]]
After PCA being applied, the features used for training are listed below:
      PCA_1    PCA_2    PCA_3    PCA_4

[160 rows x 4 columns]
169   0
175   1
619   1
749   1
477   0
     ..
174   0
623   0
235   1
368   1
831   1
Name: label, Length: 160, dtype: int64
```

Step 5:

Initialization of individual client models which are designated as quantum clients is done.

Step 6:

Creation of global model with the aggregation of both quantum and classical models.

Step 7:

Application federated learning round is done for both classical and quantum clients. Then aggregation is being done with ‘Federated Averaging’.

Step 8:

The experimentation is done for 3 rounds of federated learning. During this, the quantum clients 0 and 1 are trained using QNN model and the other three clients 1, 2 and 3 are trained with classical ML model.

```
Federated Round 1
Training Client 0 (Quantum)
Training Client 1 (Quantum)
Training Client 2 (Classical)
Training Client 3 (Classical)
Training Client 4 (Classical)
Federated Round 2
Training Client 0 (Quantum)
Training Client 1 (Quantum)
Training Client 2 (Classical)
```

```
Training Client 3 (Classical)
Training Client 4 (Classical)
Federated Round 3
Training Client 0 (Quantum)
Training Client 1 (Quantum)
Training Client 2 (Classical)
Training Client 3 (Classical)
Training Client 4 (Classical)
```

Step 9:

Aggregation of all the clients training into the global model.

Result:

Aggregating from 3 compatible clients.

Step 10:

Compute the loss and accuracy of final global model after individual client models aggregation.

```
Completed Federated Round 1
Completed Federated Round 2
Completed Federated Round 3
Completed Federated Round 4
Completed Federated Round 5
Completed Federated Round 6
Completed Federated Round 7
Completed Federated Round 8
Completed Federated Round 9
Completed Federated Round 10
.
Completed Federated Round 50
Global Model Accuracy after Federated Learning: 0.5964
```

4.3 Implementation of Pure Quantum Client Model

```
Completed Quantum Round 1
Completed Quantum Round 2
Completed Quantum Round 3
Completed Quantum Round 4
Completed Quantum Round 5
Completed Quantum Round 6
Completed Quantum Round 7
Completed Quantum Round 8
Completed Quantum Round 9
Completed Quantum Round 10
…...
```

```
Completed Quantum Round 50
Quantum Accuracy: 0.5006241817129982
```

4.4 Implementation of Classical ML Client Model

```
Completed Classical FL Round 1
Completed Classical FL Round 2
Completed Classical FL Round 3
Completed Classical FL Round 4
Completed Classical FL Round 5
Completed Classical FL Round 6
Completed Classical FL Round 7
Completed Classical FL Round 8
Completed Classical FL Round 9
Completed Classical FL Round 10
:
Completed Classical FL Round 50
Classical FL Accuracy: 0.5004719422708035
```

5. Comparative Analysis

The proposed work compared the performance of the client-side models executions with variational parameters set-up such as clients with hybrid ML models having both classical and quantum, pure quantum and pure classical ML models. The accuracy of these models are shown in figure 2.

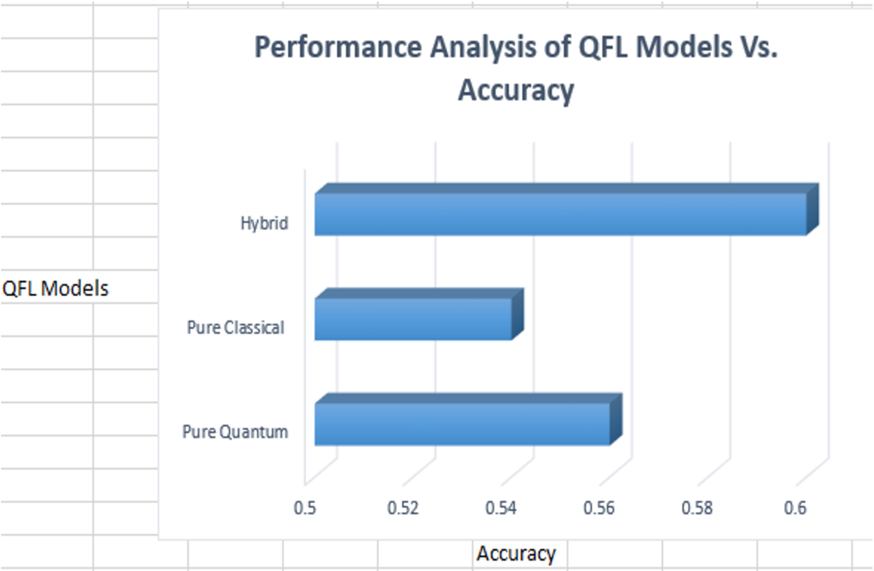

Fig.2. Performance Analysis of QFL Models at Client side with variational parameter set-up

6. Results and Discussion

The experimentation results show that, in the conventional system configuration set-up, the QFL under hybrid (classical with Quantum) models running in clients has given the accuracy of 60% when compared to other variational parameter set-ups with clients having pure Quantum ML models or classical ML models exhibited around 50% accuracy even after fine-tuning the hyper-parameters.

However, if the number of rounds are increased and all the clients are quantum clients and the experimentation is done in QPU or Quantum computers, then the accuracy will be dramatically increased. Also, the training loss associated with each individual client is high which indicates, it needs to be stabilized by increasing the number of rounds as typical FL rounds will be 20 to more than 200 and the number of layers with appropriate changes in the hyper-parameters. As the quantum model weights are based on variational circuits and the classical models are weighted using NN weights, their loss landscape will be impacting a huge as they both are completely different from each other. Hence, it is identified that, in the future work, these insights will be used to improve the global model accuracy.

7. Conclusion and Future Work

This paper investigated the applicability of Quantum Federated Learning (QFL) for real-time healthcare analytics by integrating Quantum machine learning models in Federated learning paradigm. By combining federated learning with quantum computing, QFL aims to enhance the distributed model training while maintaining model privacy. In this proposed research work, the framework was implemented in a conventional classical system configuration where quantum models are simulated rather than executed on quantum computer. The experimental results showed a hybrid configuration where clients used both classical and quantum ML models achieved an accuracy of 60% outperforming the configurations where clients used purely classical or purely quantum models which achieved nearly 50% accuracy even after parameter fine-tuning. However, the training loss observed across clients remained relatively high, indicating the instability in the model’s training process. This can be attributed to the fundamental differences between classical neural network weight optimization and variational quantum circuit parameter optimization.

Future work will focus on improving the stability and performance of QFL framework by increasing the number of federated rounds to enhance the coverage and enhancing deeper variational parameters optimization with improved hyper-parameter tuning. Further, this can be experimented on real QPU systems and cloud-based quantum computers so that, the real benefit of QFL can be received.

References:

[1] C. Ren *et al.*, "Toward Quantum Federated Learning," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 9, pp. 15580-15600, Sept. 2025, doi: 10.1109/TNNLS.2025.3552643, (2025).

[2] K.Yu, F.Gao & S.Lin, Quantum federated learning for distributed quantum networks. arXiv preprint arXiv:2212.12913, https://doi.org/10.48550/arXiv.2212.12913, (2022).

[3] W. Li & D.L.Deng, Quantum delegated and federated learning via quantum homomorphic encryption. Research Directions: Quantum Technologies, 3, e3, https://doi.org/10.48550/arXiv.2409.19359, (2025).

[4] Y. Song, Y.Wu, S.Wu, D.Li, Q.Wen, S.Qin, & F.Gao, A quantum federated learning framework for classical clients. Science China Physics, Mechanics & Astronomy, 67(5), 250311, https://doi.org/10.1007/s11433-023-2337-2, (2024).

[5] R. Rofougaran, H.Yoo, H.Tseng and C. Chen, "Federated Quantum Machine Learning with Differential Privacy," *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Seoul, Korea, Republic of, 2024, pp. 9811-9815, doi: 10.1109/ICASSP48485.2024.10447155, (2024).

[6] J.Shi , T. Chen, S. Zhang & X.Li, Personalized quantum federated learning for privacy image classification. arXiv preprint arXiv:2410.02547, https://doi.org/10.48550/arXiv.2410.02547, (2024).

[7] M. Chehimi, Chen, S.YC., Saad, W. et al. Federated quantum long short-term memory (FedQLSTM). Quantum Mach. Intell. 6, 43 (2024). https://doi.org/10.1007/s42484-024-00174-z

[8] N. Innan, A. Marchisio, M. Bennai & M. Shafique, Exploring quantum federated learning: Algorithms, applications, and challenges. Quantum Computational AI, 25-50, https://doi.org/10.1016/B978-0-44-330259-6.00010-4, (2026).

[9] T. Wang, H. Tseng and S. Yoo, "Quantum Federated Learning with Quantum Networks," *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Seoul, Korea, Republic of, 2024, pp. 13401-13405, doi: 10.1109/ICASSP48485.2024.10447516, (2024).

[10] D.Gurung, & S.Pokhrel, A personalized quantum federated learning. In Proceedings of the 8th Asia-Pacific Workshop on Networking (pp. 175-176), https://doi.org/10.1145/3663408.3665806, (2024).

[11] https://data.telangana.gov.in/dataset/telangana-health-department-diagnostics-data